

## REMARKS

Claims 43-55 and 58 are now pending in this application. In the outstanding Office Action, claims 43-55 and 58 stand rejected under 35 U.S.C. § 103(a) as being allegedly unpatentable over Non-Patent Literature “The Case for Non-transparent Replication: Examples from Bayou” by Douglas B. Terry et al. (“Terry”) in view of United States Patent No. 5,630,116 (“Takaya”). Applicants respectfully traverse.

Claims 43-55 and 58 have been amended. Claims 59-64 have been added.

### *Interview Summary*

On May 14, 2008, Examiner Fred Ehichioya and Applicants’ undersigned representative, Mr. Eiferman, participated in a telephonic interview. During the interview, Mr. Eiferman proposed the claim amendments herein. Examiner Ehichioya agreed to reevaluate the pending rejections in light of the claim amendments and remarks herein.

## Claims 43-50

Claims 43-55 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Terry in view of Takaya. Claim 43 as amended recites in part a method for client masterd replication comprising creating a first replica file on a server device based upon a master file *in conformance with the at least one replication option wherein the replication indicates a portion of the master file to be replicated and wherein each replication option is associated with a security option for controlling replication of the associated portion of the master file*, applying a server masterd replication scheme to the replica file on the server with respect to each of at least one replicating client device and synchronizing the fist replica file on the master client device to the master file on the master client device using a conflict resolution scheme.

Support for this amendment can be found, for example, on pages 10-14 of the application. Briefly summarizing this discussion, server masterd replication systems have the drawbacks in that a user does now know whether local changes made at a client will be replicated at a master and that if a server reaches its full capacity, the master may be deleted

from the server with such deletion propagating to the replica at the client. File sharing systems also have associated drawbacks including tedious requirements for copying files to a sharing folder, difficulties of the user in determining which copies contain the most recent changes and achieving parity between a local folder structure and sharing folder structure.

Applicant's application describes a client mastered replication method and system that avoids the drawbacks associated with server mastered schemes and file sharing schemes and enables a user to work on the most updated version of the master at all times. The method and system also allows for the possible deletion of a replica on a server without causing propagation of the deletion to the master file.

A client stores one or more client files that may be designated as master client files in order to cause those master client files to be replicated. The client may replicate a designated master client file to a server to generate a first replica file. Changes may then be made to first replica file, for example, by having the server execute a server mastered replication system with respect to at least one replicating client. If the changes to the first replica do not conflict with local changes to the master client file as determined by applying a conflict resolution scheme, then the changes may be replicated back to the master client file.

A file system is provided at a client that operates to manage the organization and storage of master files and enables files to be selected for replication. Files selected for replication as master files may be selected, for example, by opening a file and selecting a replication property to deem the file a master file. In addition, files may have associated replication options, which may be a replication level option that determines whether the entire contents of the master will be selected for replication or whether only portions of the contents will be selected. Replication may be selected, for example, on a paragraph by paragraph level, a section by section level or a page by page level.

A synchronization manager is provided at a client, which operates to identify new changes made to a master file. The synchronization manager may be configured to replicate a master file to a server in response to an event that occurs at the client. For example, a replication time interval may be selected by a user at the client and the master may be replicated at an expiration of a selected time interval, when the file is saved at the client or in response to a request from the server.

Changes made to the replica at the server need not necessarily be propagated to the client. Thus, if the server reaches its storage capacity, the replica may be deleted from the server without deleting the master from the client. The client may provide a replication notification to alert a user when the master is replicated or deleted from the server. Such a notification may be a user interface, for example, an icon overlay in a file system. When the master is replicated to the server the icon may be displayed. The icon may remain displayed as long as the replica is stored on the server. However, if the server reaches its storage capacity and the replica is deleted from the server, the icon may be removed. A message may be displayed asking whether the user wishes to again replicate the master to the server or wishes to replicate the master to another computing device.

One or more replicating clients may also participate in the replication scheme. Each replicating client replicates the first replica from the server to create a second replica. Local changes made to the second replica at the replication client may be replicated back to the first replica at the server. In particular, due to the fact that conflicts may occur when the second replicas are replicated back to the first replica, the server may replicate the first replica to its connected replication clients in accordance with a server mastered replication method. To implement such a method, the first replica becomes a master file with each second replica becoming a direct replica. Although the first replica may become a master file with respect to each second replica, the first replica nonetheless remains a replica with respect to the client master file at the client.

In particular, changes made at the replicating clients may be propagated to the first replica file at the server in accordance with the sever based replication method. The first replica having incorporated changes from the replicating files in accordance with the conflict resolution scheme may then be replicated back to the master client file in accordance with a conflict resolution scheme.

The Examiner asserts that Terry discloses a method of client mastered replication, utilizing an exemplary scenario presented on page 3 of Terry. This exemplary scenario contemplates an exemplary user Alice managing a personal calendar. According to the example, Alice maintains a replica of her calendar on her office machine, one on her laptop and also one on the office machine of her administrative assistant Bob. Alice and Bob's office machines perform reconciliation with each other on a frequent basis so that any

updates made to the calendar by either Alice or Bob are seen by the other. When Alice is traveling she may update the replica on her laptop while the laptop is disconnected. Alice occasionally connects to her or Bob's office machine to exchange recently added meetings thereby updating their replicas of the shared calendar.

The Examiner interprets Alice's laptop as the master client computing device, Alice's office machine as the server on which the first replica is stored and Bob's office machine as the replicating computing device that receives a copy of a first replica.

The Examiner admits that Terry does not explicitly disclose a master file and instead relies on Takaya for this claim limitation. Takaya relates to an automatic delivery system for master files in a distributed system that includes a parent workstation, a master file updating section for updating master files, a delivery start notifying section for forwarding the directory list of master files to a child workstation and a master delivery section for forwarding data or information requested by the child workstation. The child workstation has a slave file updating section for updating slave files, an updated asset registering section for forwarding updated data to the master files and a slave delivery section for comparing the contents of the directory list of the master files and those of the directory list of the slave files and requesting the parent workstation to forward the data or information, which is found different.

However, neither Terry nor Takaya taken alone or in combination teaches or suggests creating a first replica file on a server device based upon a master file, applying a server mastered replication scheme to the replica file on the server with respect to each of at least one replicating client device and synchronizing the fist replica file on the master client device to the master file on the master client device using a conflict resolution scheme. In particular, if Alice's laptop is the master client device, her office machine is the server and Bob's office machine is the replicating client, there is no disclosure that Alice's office machine and Bob's office machine execute a server mastered replication scheme. In particular, the deletion of a file on Alice's office machine could result in the deletion of the master file on Alice's laptop.

Moreover, neither Terry nor Takaya alone or in combination discloses a replication property, let alone a replication option *wherein the replication indicates a portion of the master file to be replicated and wherein each replication option is associated with a security option for controlling replication of the associated portion of the master file*. As previously

noted, files may be replicated on a paragraph by paragraph, section by section or using some other portion definition.

In light of the fact that the cited references fail to teach or suggest the recited limitations, claim 43 should be allowed. Claims 44-50 depend from claim 43 and therefore include all the limitations of claim 43. Therefore, for the reasons stated above with respect to claim 43, claims 44-50 should be allowed.

### **Claims 51-55 and 58**

Claim 51 as amended includes limitations similar to claim 43. Thus, for at least the reasons stated with respect to claim 43, claim 51 should be allowed. Claims 52-55 and 58 depend from and include all the limitations of claim. Therefore, for at least the reasons stated with respect to claim 51, claims 52-55 and 58 should be allowed.

**DOCKET NO.:** 302762.01 / MSFT-2782  
**Application No.:** 10/722,247  
**Office Action Dated:** March 11, 2008

**PATENT**

### **CONCLUSION**

In view of the above amendments and remarks, applicant respectfully submits that the present invention is in condition for allowance. Reconsideration of the application is respectfully requested.

Date: July 11, 2008

/Kenneth R. Eiferman/  
Kenneth R. Eiferman  
Registration No. 51,647

Woodcock Washburn LLP  
Cira Centre  
2929 Arch Street, 12th Floor  
Philadelphia, PA 19104-2891  
Telephone: (215) 568-3100  
Facsimile: (215) 568-3439